

УДК 004.3

А. А. Олейник, канд.техн.наук, доц.,  
С. Ю. Скрупский, канд.техн.наук, ст.преп.,  
С. А. Субботин, канд.техн.наук, доц.Запорожский национальный технический университет  
olejnikaa@gmail.com, sskrupsky@gmail.com, subbotin@zntu.edu.ua

## Модель параметрического синтеза нейро-нечетких сетей в ярусно-параллельной форме

*Рассмотрена задача автоматизации параметрического синтеза нейро-нечетких сетей по заданным наборам наблюдений. Предложена модель в ярусно-параллельной форме обучения нейро-нечетких сетей, которая основывается на вероятностном подходе при поиске значений настраиваемых параметров и заключается в распределении наиболее ресурсоемких этапов по узлам параллельной вычислительной системы. Проведены эксперименты по решению практических задач диагностирования на основе предложенной модели.*

**Ключевые слова:** параметрический синтез, нейро-нечеткие сети, случайный поиск, параллельная система, ускорение.

### Введение

При построении интеллектуальных систем распознавания образов, контроля качества и прогнозирования [1–3] для моделирования исследуемых объектов и процессов широко применяются искусственные нейро-нечеткие сети [4–7]. Синтез таких сетей, как правило, осуществляется на основе градиентных методов обучения [5–8], использование которых на практике связано с проблемами выбора начальной точки поиска, необходимостью вычисления производных целевой функции, а также с низкой сходимостью. Поэтому для параметрического синтеза нейро-нечетких сетей целесообразно использовать стохастические методы [9–12], не требующие вычисления производных и позволяющие решать различные задачи оптимизации, возникающие при построении гибридных моделей вычислительного интеллекта. Однако такие методы в силу использования вероятностного подхода обладают низкой скоростью сходимости, что обуславливает целесообразность разработки моделей и методов, позволяющих выполнять случайный поиск оптимальных значений параметров нейро-нечетких сетей на различных узлах параллельной системы.

Целью работы является создание модели параметрического синтеза нейро-нечетких сетей в ярусно-параллельной форме.

### Постановка задачи параметрического синтеза нейро-нечетких сетей

Пусть задан набор объектов  $S = \langle P, T \rangle$ , где  $P$  – множество характеристик объектов;  $T$  – множество значений отклика. Множества значений  $P$  и  $T$  представляются, соответственно,

в виде матрицы  $P = (p_{qm})_{QM}$  и вектора  $T = (t_q)_Q$ ,

где  $p_{qm}$  – значение  $m$ -го атрибута  $q$ -го объекта ( $m = 1, 2, \dots, M$ ,  $q = 1, 2, \dots, Q$ );  $t_q$  – значение отклика  $q$ -го объекта;  $M$  – число атрибутов;  $Q$  – число объектов.

Тогда задача параметрического синтеза нейро-нечеткой модели  $NFN$  заключается в идентификации ее параметров таким образом, чтобы обеспечивалось приемлемое значение заданного критерия качества  $G$ . В качестве целевого критерия  $G$  при обучении нейро-нечетких моделей могут быть использованы, например, ошибка распознавания или среднеквадратическая ошибка.

### Модель процесса параметрического синтеза нейро-нечетких сетей в ярусно-параллельной форме

Как отмечено выше, для обучения нейро-нечетких сетей эффективно могут применяться методы случайного поиска с адаптацией (методы генетического, эволюционного, стохастического поиска) [9–12].

Однако для получения приемлемых результатов (моделей, обеспечивающих приемлемую точность распознавания или прогнозирования) однократного запуска таких методов, как правило, недостаточно, что обусловлено использованием вероятностного подхода и закливанием в некоторых случаях в локальных оптимумах.

Необходимость многократного запуска, а также существенное время поиска обуславливают целесообразность распараллеливания процесса обучения гибридных интеллектуальных моделей, в том числе и нейро-нечетких сетей.

С целью разработки параллельного метода случайного поиска для обучения нейро-нечетких моделей представим процесс параметрического синтеза в ярусно-параллельной форме. Для этого выделим этапы случайного поиска, которые рационально распараллелить. Как известно [9–12], основными этапами вероятностных методов являются:

– инициализация начального множества решений  $R^{(0)}$  (1):

$$R^{(0)} = \{\chi_1^{(0)}, \chi_2^{(0)}, \dots, \chi_{N_\chi}^{(0)}\}, \quad (1)$$

где  $\chi_k^{(0)} = \{g_{1k}^{(0)}, g_{2k}^{(0)}, \dots, g_{N_g k}^{(0)}\}$  –  $k$ -е решение во множестве  $R^{(0)}$ ,  $k = 1, 2, \dots, N_\chi$ ;

$N_\chi$  – число элементов множества  $R^{(0)}$  (число случайно сгенерированных решений при инициализации), как правило, величина  $N_\chi$  не меняется в процессе вероятностной оптимизации;  $g_{lk}^{(0)}$  – значение  $l$ -го элемента (параметра) в  $k$ -м решении,  $l = 1, 2, \dots, N_g$ ;

$N_g$  – число параметров в решении  $\chi_k^{(i)}$ ;

– оценивание текущего ( $i$ -го) множества решений  $R^{(i)}$  (2):

$$G^{(i)} = G(R^{(i)}), \quad (2)$$

где  $G$  – целевая функция, позволяющая оценить качество  $i$ -го множества решений  $R^{(i)}$ ;

– проверка критериев останова;  
– формирование нового множества решений  $R^{(i+1)}$ . При использовании эволюционного поиска данный этап выполняется посредством применения операторов скрещивания и мутации.

Для сокращения времени случайного поиска при построении гибридных моделей вычислительного интеллекта целесообразно на этапе инициализации при создании начального множества решений использовать априорную информацию об обучающей выборке. Следовательно, при таком подходе потребуются  $N_\chi$  раз синтезировать гибридные модели (сформировать множества начальных значений их параметров), что требует значительных компьютерных ресурсов и обуславливает необходимость распараллеливания данного этапа.

Наиболее ресурсоемким, как правило, является этап оценивания текущего множества решений  $R^{(i)}$ , на котором затрачивается большое количество компьютерных и временных ресурсов при вычислении значений целевой функции  $G$  для каждого  $k$ -го ( $k = 1, 2, \dots, N_\chi$ )

решения  $\chi_k^{(i)}$ :  $G_k^{(i)} = G(\chi_k^{(i)})$ . Поскольку данный этап обладает высокой вычислительной сложностью, выполняется медленно и не требует обмена данными между решениями  $\chi_k^{(i)}$ , его рационально выполнять параллельно.

Важным этапом является создание нового множества решений  $R^{(i+1)}$ . С целью более детального исследования областей локальных оптимумов целесообразным является разбиение текущего множества решений  $R^{(i)}$  на подмножества  $R^{(i,j)}$  с последующим поиском оптимума в каждом из них (3):

$$R^{(i)} \rightarrow \{R^{(i,1)}, R^{(i,2)}, \dots, R^{(i,N_{pr})}\}, \quad (3)$$

где  $R^{(i,j)}$  –  $j$ -е подмножество  $i$ -го множества  $R^{(i)}$ ;  $N_{pr}$  – количество процессов, одновременно выполняемых в параллельной компьютерной системе.

Разбиение (3) предлагается выполнять с учетом априорной информации о расположении решений  $\chi_k^{(i)}$  в пространстве элементов  $g_l$  ( $l = 1, 2, \dots, N_g$ ). Такой подход, в отличие от применения островной модели эволюционного поиска [9, 12], подразумевающей случайное формирование подпопуляций  $R^{(i,j)}$ , позволяет учитывать информацию о пространственном расположении решений  $\chi_k^{(i)}$  в множестве  $R^{(i)}$  и более детально исследовать области возможных оптимумов. Количество решений  $|R^{(i,j)}|$  в каждом подмножестве  $R^{(i,j)}$  определяется как отношение общего числа решений  $N_\chi$  в множестве  $R^{(i)}$  к количеству процессов  $N_{pr}$  (4):

$$|R^{(i,j)}| = N_\chi / N_{pr}. \quad (4)$$

Таким образом, задача разбиения (3) сводится к необходимости формирования  $N_{pr}$  подмножеств  $R^{(i,j)}$  (каждое из которых состоит из  $|R^{(i,j)}|$  решений) в заданном пространстве элементов  $g_l$  ( $l = 1, 2, \dots, N_g$ ) (5):

$$R^{(i,j)} = \left\{ \chi_1^{(i,j)}, \chi_2^{(i,j)}, \dots, \chi_{|R^{(i,j)}|}^{(i,j)} \right\}, \quad j = 1, 2, \dots, N_{pr}. \quad (5)$$

Случайный поиск с адаптацией в каждом  $j$ -м подмножестве  $R^{(i,j)}$  предлагается выполнять на  $j$ -м процессе параллельной компьютерной системы ( $j = 1, 2, \dots, N_{pr}$ ) на протяжении  $N_{it}$  итераций.

После выполнения  $N_{it}$  итераций

случайного поиска над каждым подмножеством  $R^{(i,j)}$  выполняется их объединение  $R^{(i)} = R^{(i,1)} \cup R^{(i,2)} \cup \dots \cup R^{(i,N_{pr})}$  в единую популяцию с дальнейшим проведением вероятностной оптимизации над объединенным множеством. Это позволит выявлять новые области, содержащие локальные (возможно, и глобальный) оптимумы. С целью сокращения времени вероятностной оптимизации при работе с объединенным множеством решений предлагается вычисление значений целевой функции  $G_k^{(i)}$  решений  $\chi_k^{(i)}$  выполнять на процессах в параллельной системе (распараллелить).

Исходя из отмеченного выше, представим граф процесса параметрического синтеза нейро-нечетких моделей в ярусно-параллельной форме (рис. 1). Начиная с яруса проверки критериев останова процесс выполняется итеративно.

На рис. 1 операция  $RS(R)$  представляет собой случайный поиск в множестве решений  $R$ . Последняя операция на графе –  $RS(R^{(i)})$  выполняется параллельно. Одна итерация процесса случайного поиска  $RS(R^{(i)})$  в ярусно-параллельной форме представлена на рис. 2

Пунктирные линии на рис. 1 и 2 отделяют ярусы графа друг от друга. Операции на одном ярусе графа выполняются параллельно. Дуги графа показывают зависимости между операциями. Так, нельзя приступить к выполнению любой операции пока не завершатся другие операции, дуги из которых ориентированы в вершины рассматриваемых операций.

Переход между ярусами, при котором от одной вершины графа дуги ориентированы в несколько вершин следующего яруса, сопряжен с рассылкой (Scatter) решений по процессам. Переход другого типа, при котором дуги ориентированы от нескольких вершин одного яруса к одной вершине следующего яруса, выполняется при помощи сбора (Gather) решений в главном процессе.

Как видно из рис. 1 и 2, наиболее ресурсоемкие операции, а именно инициализация начального множества решений, оценивание текущего множества решений, случайный поиск в подмножестве решений, выполняются одновременно на процессах параллельной системы.

Предложенная модель параметрического

синтеза нейро-нечетких сетей в ярусно-параллельной форме основывается на вероятностном подходе при поиске значений настраиваемых параметров и заключается в распределении наиболее ресурсоемких этапов по узлам параллельной вычислительной системы, что позволяет сократить время настройки параметров (значений весовых коэффициентов и параметров функций принадлежности нейроэлементов) синтезируемых нейромоделей.

### Эксперименты и результаты

Для проведения экспериментального исследования использовался кластер Института проблем моделирования в энергетике имени Г.Е. Пухова НАН Украины. Было задействовано 16 логических узлов, каждый из которых выполнял один процесс, следующей конфигурации: процессор – Intel Xeon 5405, оперативная память – 4×2 Гб DDR-2 на каждый вычислительный узел, коммуникационная среда InfiniBand 20Гб/с. На узлах кластера установлено промежуточное программное обеспечение – планировщик Torque, пакет OMPI.

Для оценивания эффективности предложенной модели на ее основе был разработан и программно реализован параллельный эволюционный метод. Программное обеспечение разработано на языке C++ с применением библиотеки MPI [13]. Измерения времени были реализованы при помощи библиотечной функции MPI\_Wtime. Эксперименты были выполнены по 5 раз на одном, двух, четырех, шести, восьми, десяти, двенадцати, четырнадцати и шестнадцати процессах.

С помощью разработанной модели и реализованного на основе ее метода и программного обеспечения выполнялось решение задачи диагностирования лопаток газотурбинных авиадвигателей [14], выявление неисправности которых на ранних стадиях является важной и актуальной задачей, решение которой позволит повысить надежность и долговечность двигателей летательных аппаратов. Задача заключалась в построении диагностической модели, позволяющей выполнять распознавание некондиционных лопаток авиадвигателей по значениям спектров мощности свободных затухающих колебаний после ударного возбуждения.

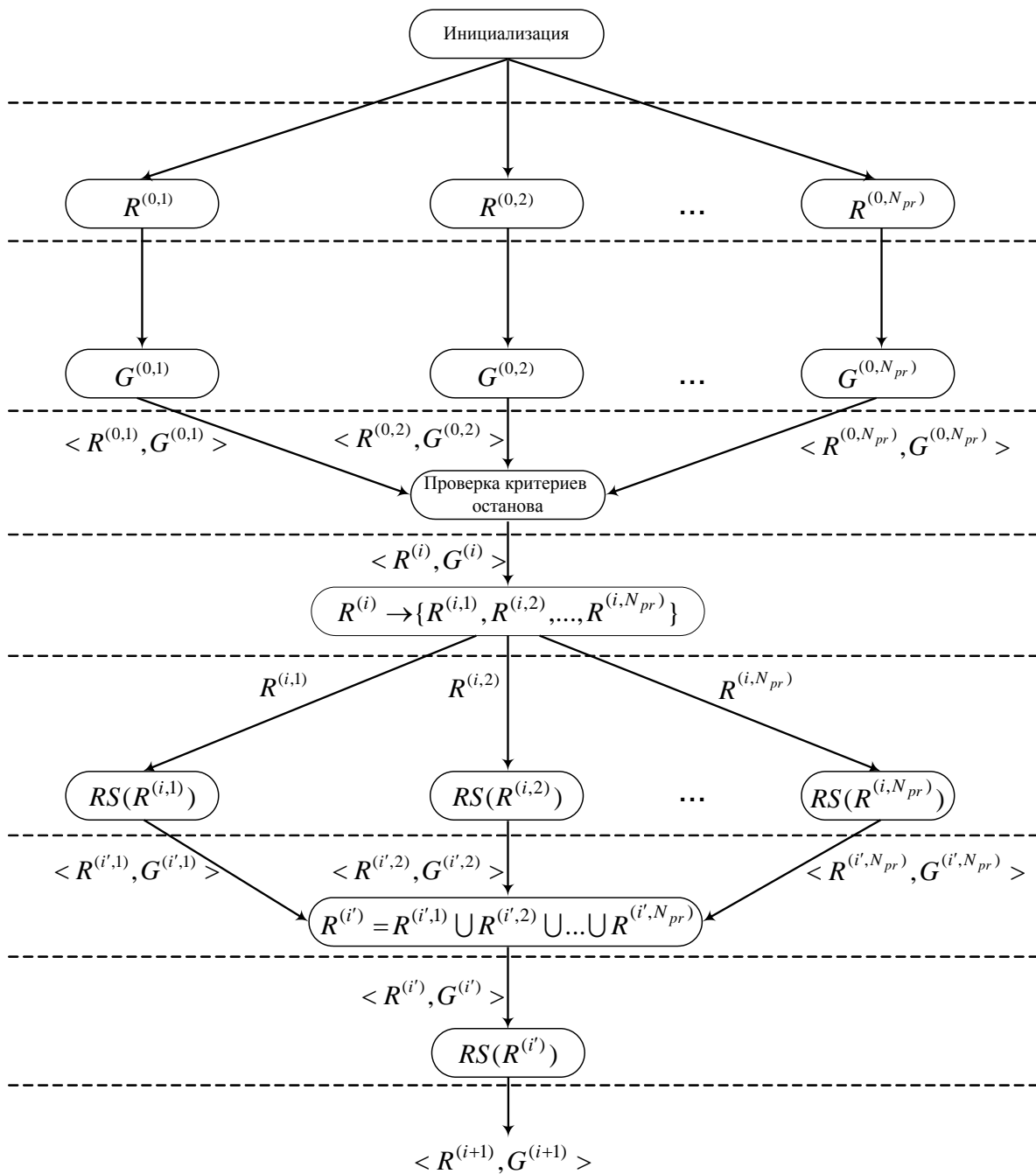


Рисунок 1 – Граф процесса параметрического синтеза нейро-нечетких сетей в ярусно-параллельной форме

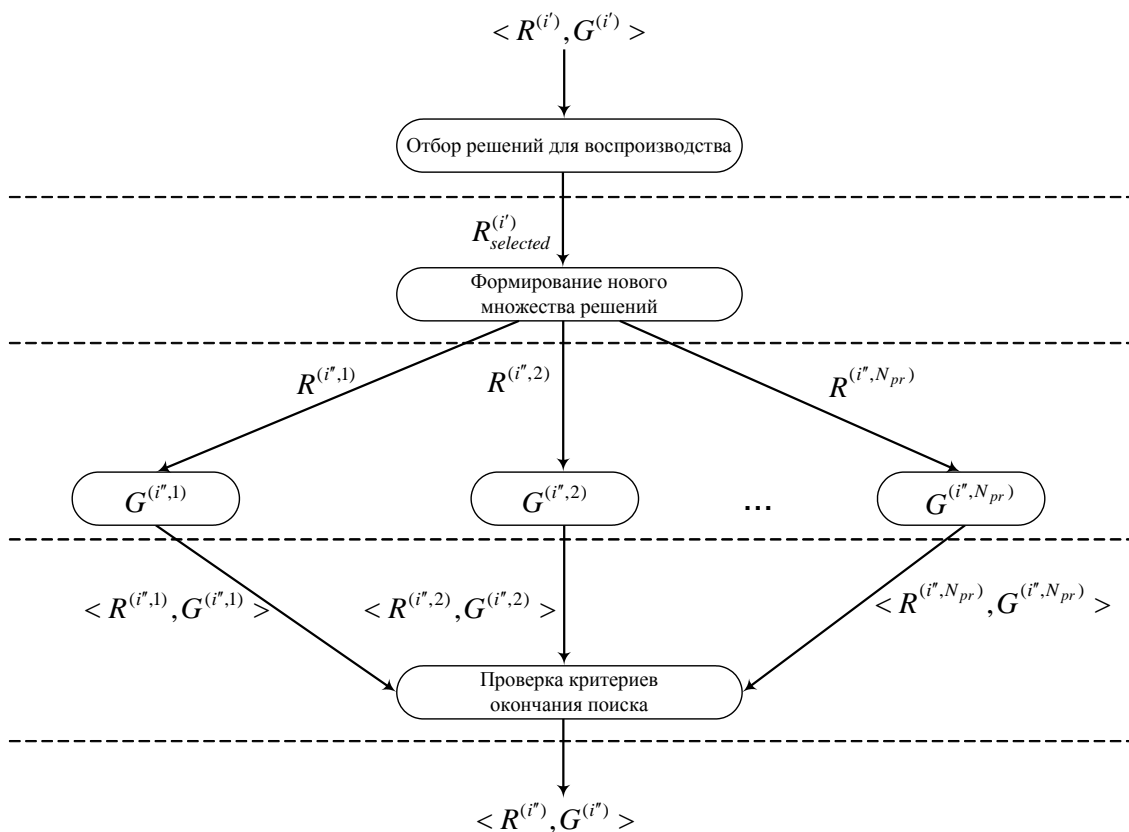


Рисунок 2 – Граф одной итерации случайного поиска  $RS(R^{(i')})$

На рис. 3 приведены усредненные временные затраты на построение нейро-нечетких моделей с помощью метода, реализующего предложенную модель, в параллельной системе.

В табл. 1 приведены показатели эффективности кластера, выполняющего

параметрический синтез нейро-нечетких сетей предложенным методом.

На рис. 4 и 5 приведены графики ускорения и эффективности параллельной системы при выполнении метода, реализующего предложенную модель, соответственно.

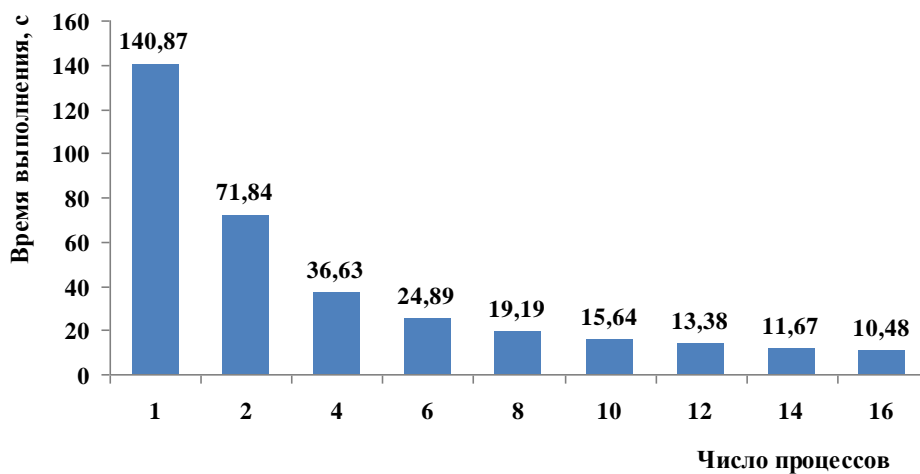


Рисунок 3 – Усредненные временные затраты на выполнение метода, реализующего предложенную модель, в параллельной системе

Таблица 1. Показатели эффективности кластера, выполняющего метод, который реализует предложенную модель

Число процессов	Время выполнения, с	Ускорение	Эффективность
1	140,87	1	1
2	71,84	1,96	0,98
4	36,63	3,85	0,96
6	24,89	5,66	0,94
8	19,19	7,34	0,92
10	15,64	9,01	0,90
12	13,38	10,53	0,88
14	11,67	12,07	0,86
16	10,48	13,45	0,84

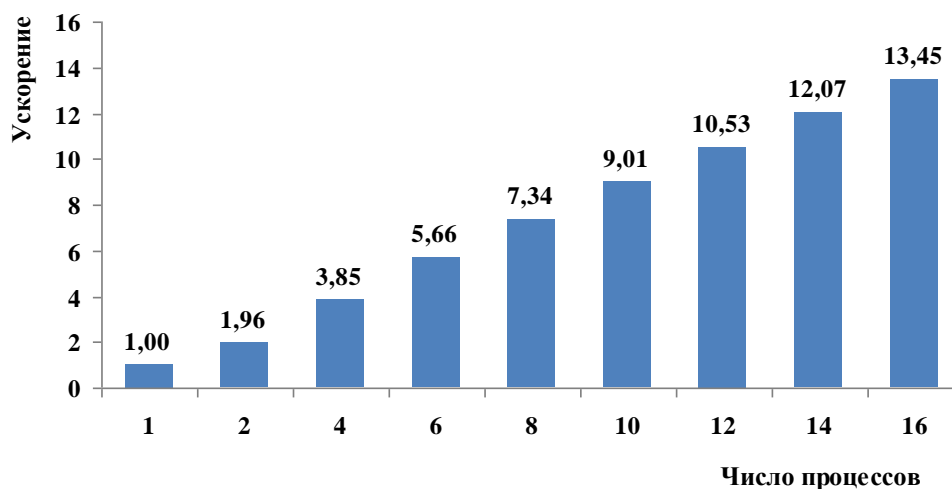


Рисунок 4 – График ускорения процесса параметрического синтеза нейро-нечетких сетей

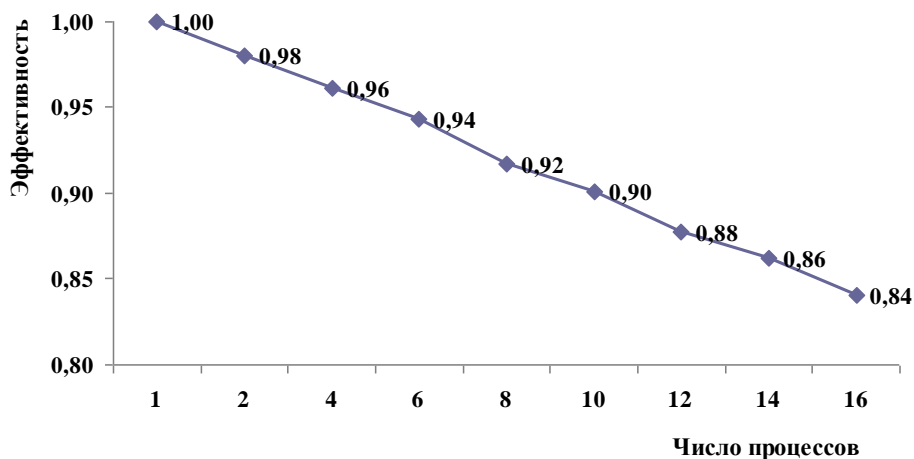


Рисунок 5 – График эффективности параллельной системы, выполняющей параметрический синтез нейро-нечетких сетей

Как видно из графиков, приведенных на рис. 4 и 5, ускорение почти линейно растет пропорционально числу процессов. При этом медленное снижение эффективности системы с ростом числа процессов свидетельствует о хорошей масштабируемости метода, реализующего предложенную модель на параллельной архитектуре MPP [13].

Главной причиной снижения ускорения являются пересылки. Для оценки доли

пересылок между процессами были реализованы вызовы функций MPI\_Wtime непосредственно перед и после каждой пересылки. Так было вычислено суммарное время пересылок. Результаты этой оценки приведены в табл. 2 и на рис. 6. Доля пересылок растет быстрее с увеличением числа процессов. Это показывает, что начиная, примерно, с 42х процессов, доля пересылок превысит долю целевых вычислений, и эффективность такой системы будет низкой..

Таблиця 2. Доля пересылок между процессами

Число процессов	Время пересылок, с	Доля пересылок
1	0	0
2	1,44	0,02
4	1,47	0,04
6	1,49	0,06
8	1,73	0,09
10	1,72	0,11
12	1,87	0,14
14	1,87	0,16
16	1,99	0,19

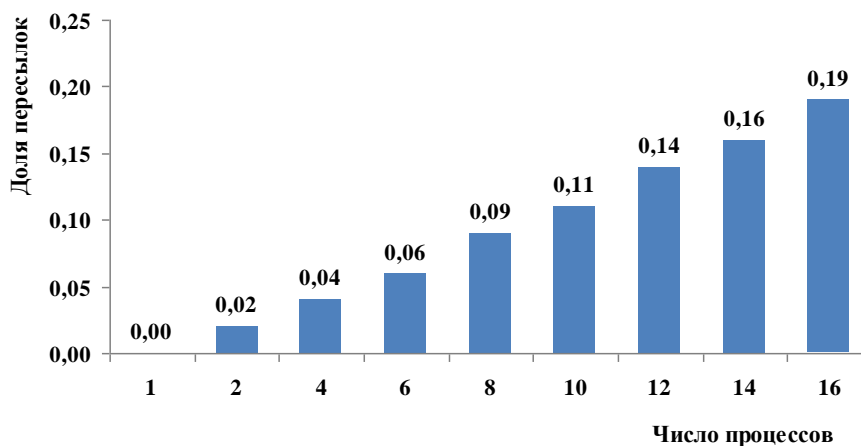


Рисунок 6 – Зависимость доли пересылок между процессами, реализующими параметрический синтез нейро-нечетких сетей в соответствии с предложенной моделью, от числа процессов

Таким образом, проведенные на кластере эксперименты свидетельствуют о приемлемых значениях показателей эффективности параллельного метода, реализующего предложенную модель параметрического синтеза нейро-нечетких сетей. Результаты экспериментов по исследованию зависимости показателей эффективности и доли пересылок от числа процессов показали, что на практике предложенную модель и разработанное программное обеспечение целесообразно применять в кластерах типа MPP с числом задействованных процессов до 42х, что позволяет получить результаты за приемлемое время.

### Выводы

В работе решена актуальная задача автоматизации моделирования процесса параметрического синтеза нейро-нечетких сетей по заданным наборам прецедентов.

Научная новизна работы заключается в том, что предложена модель параметрического синтеза нейро-нечетких сетей в ярусно-

параллельной форме, которая основывается на вероятностном подходе при поиске значений настраиваемых параметров и заключается в распределении наиболее ресурсоемких этапов по узлам параллельной вычислительной системы, что позволяет сократить время настройки параметров (значений весовых коэффициентов и параметров функций принадлежности нейроэлементов) синтезируемых нейромоделей.

Практическая ценность полученных результатов заключается в том, что разработано программное обеспечение, позволяющее выполнять параметрический синтез нейро-нечетких сетей на основе параллельного случайного поиска.

Работа выполнена в рамках государственной научно-исследовательской темы Запорожского национального технического университета «Интеллектуальные информационные технологии автоматизации проектирования, моделирования, управления и диагностирования производственных процессов и систем» (номер государственной регистрации 0112U005350).

**Список литературы**

1. Sobhani-Tehrani E. Fault diagnosis of nonlinear systems using a hybrid approach / E. Sobhani-Tehrani, K. Khorasani. – New York: Springer, 2009. – 265 p. – (Lecture notes in control and information sciences ; № 383).
2. Price C. Computer based diagnostic systems / C. Price. – London: Springer, 1999. – 136 p.
3. Denton T. Advanced automotive fault diagnosis / T. Denton. – London: Elsevier, 2006. – 271 p.
4. Бодяньський Є. В. Нейро-фаззи моделі в системах штучного інтелекту : навч. посібник / Є. В. Бодяньський, Є. І. Кучеренко. – Харків: ХНУРЕ, 2006. – 196 с.
5. Jang J. R. ANFIS: Adaptive-network-based fuzzy inference system / J. R. Jang // IEEE transactions on systems and cybernetics. – 1993. – Vol. 23. – P. 665–685.
6. Rutkowski L. Flexible neuro-fuzzy systems : structures, learning and performance evaluation / L. Rutkowski. – Boston: Kluwer, 2004. – 276 p.
7. Бодяньський Е. В. Нейро-фаззи сети Петри в задачах моделирования сложных систем / Е. В. Бодяньський, Е. И. Кучеренко, А. И. Михалев. – Днепропетровск: Системные технологии. – 2005. – 311 с.
8. Ravindran A. Engineering optimization: methods and applications / A. Ravindran, K. M. Ragsdell, G. V. Reklaitis. – New Jersey: John Wiley & Sons, 2006. – 688 p.
9. Скобцов Ю. А. Основы эволюционных вычислений / Ю. А. Скобцов. – Донецк: ДонНТУ, 2008. – 330 с.
10. Abraham A. Engineering Evolutionary Intelligent Systems / A. Abraham, C. Grosan, W. Pedrycz. – Berlin: Springer, 2008. – 444 p.
11. Holland J. H. Adaptation in natural and artificial systems / J. H. Holland. – Ann Arbor: The University of Michigan Press, 1975. – 97 p.
12. Tenne Y. Computational Intelligence in Expensive Optimization Problems / Y. Tenne, C.-K. Goh. – Berlin: Springer: 2010. – 800 p.
13. Quinn M.J. Parallel Programming in C with MPI and OpenMP / M.J. Quinn. – New York, NY: McGraw-Hill, 2004. – 529 p.
14. Прогрессивные технологии моделирования, оптимизации и интеллектуальной автоматизации этапов жизненного цикла авиационных двигателей : монография / [А. В. Богуслаев, Ал. А. Олейник, Ан. А. Олейник, Д. В. Павленко, С. А. Субботин] ; под ред. Д. В. Павленко, С. А. Субботина. – Запорожье: ОАО "Мотор Сич", 2009. – 468 с.

Надійшла до редакції 20.01.2014

**А.О.ОЛІЙНИК, С.Ю.СКРУПСЬКИЙ, С.О.СУБОТІН**

Запорізький національний технічний університет

**МОДЕЛЬ ПАРАМЕТРИЧНОГО СИНТЕЗУ НЕЙРО-НЕЧІТКИХ МЕРЕЖ У ЯРУСНО-ПАРАЛЕЛЬНІЙ ФОРМІ**

Розглянуто задачу автоматизації параметричного синтезу нейро-нечітких мереж по заданих наборах спостережень. Запропоновано модель в ярусно-паралельній формі навчання нейро-нечітких мереж, яка ґрунтується на імовірнісному підході при пошуку значень корегуємих параметрів і полягає у розподілі найбільш ресурсоемних етапів по вузлах паралельної обчислювальної системи. Проведено експерименти з вирішення практичних завдань діагностування на основі запропонованої моделі.

**Ключові слова:** параметричний синтез, нейро-нечіткі мережі, випадковий пошук, паралельна система, прискорення.

**A.O. OLINYK, S.Yu. SKRUPSKY, S.O. SUBBOTIN**

Zaporizhzhia National Technical University

**A MODEL FOR PARAMETRICAL SYNTHESIS OF NEURO-FUZZY NETWORKS IN TIER-PARALLEL FORM**

The problem of automation of parametric synthesis of neuro-fuzzy networks for a given set of observations have been considered. A model in the tier-parallel form of neuro-fuzzy networks learning, based on a stochastic approach to locate values of adjustable parameters have been proposed. It utilizes the distribution of the most demanding stages to the nodes of parallel computer system. Experiments of the practical diagnosis problems solution based on the proposed model have been performed.

**Key words:** parametric synthesis, neuro-fuzzy networks, stochastic search, parallel system, speedup.